SEPARACIÓN CIEGA DE FUENTES

Irene Aldecoa Bilbao[1], Noemí Carné Serrano[2], Enric Monte Moreno[3]

[1,2] Estudiantes de la E.T.S. Enginyeria de Telecomunicació de Barcelona (UPC)
[3] Profesor Titular de la E.T.S. Enginyeria de Telecomunicació de Barcelona (UPC)
Departamento de Teoría de Señal y Comunicaciones, Grupo de Procesado de Voz

{alirene, alnoemi, enric}@gps.tsc.upc.es

Resumen- **La separación ciega de fuentes es un problema que consiste en obtener señales procedentes de n fuentes a partir de las mezclas que llegan a m sensores, disponiendo sólo de estas últimas. En éste artículo presentamos algoritmos de separación de señales de voz. En primer lugar resumiremos los fundamentos teóricos de los métodos de separación ciega de fuentes, y luego procederemos a la deducción y simulación de dos algoritmos de gradiente basados en el método denominado Independent Component Analysis (ICA), que realizan la separación en los dominios temporal y frecuencial.**

1. INTRODUCCIÓN

El problema de la separación de fuentes aparece en diversos campos, principalmente en procesado de señal de audio. También se aplica en arrays de antenas, sensores químicos y en geología. En todos los casos se deben separar varias fuentes linealmente superpuestas, o mezcladas, y captadas por varios sensores. Dado que los datos tienen una estructura lineal, la dificultad reside en estimar la ***matriz de mezcla***, es decir el conjunto de coeficientes en la superposición lineal, desconocida y que refleja la geometría del problema. Éste consiste en separar señales de las que a lo sumo tenemos alguna información sobre su estadística, y que han sido mezcladas en una configuración geométrica que desconocemos. De ahí el nombre de ***separación ciega de fuentes.***

En audio, se presenta el problema denominado *cocktail party* [1, 2] que se da cuando varios locutores hablan simultáneamente. En ese contexto el problema reside en centrar la atención en un único locutor de entre un número de conversaciones, y el ruido de fondo, y extraer sólo una voz. Este proceso puede modelarse como una mezcla lineal y el posterior filtrado de fuentes de sonido. El método de ***Análisis de Componentes Independientes*** (*Independent Component Analysis*, ICA) permite recuperar de forma ciega las fuentes desconocidas asumiendo que las señales originales son independientes.

2. SEPARACIÓN CIEGA DE MEZCLAS INSTANTÁNEAS CON ICA

ICA es una técnica estadística que permite hallar variables ocultas en un conjunto de medidas o señales [3, 4, 5, 6]. Dichas variables son componentes de datos estadísticos multivariables. La característica que diferencia ICA de otros métodos de separación de fuentes es que la condición para extraer componentes ocultos es que sean al mismo tiempo ***estadísticamente independientes*** y ***no gaussianos***.

2.1 Definición del modelo lineal, instantáneo y sin ruido

Sean $x_1,...,x_n$ n variables aleatorias recogidas por n sensores, que proceden de la combinación lineal de n variables aleatorias $s_1,...,s_n$, denominadas fuentes y mutuamente independientes por definición. Los ***observables*** x_i vienen dados por:

$$x_i = a_{i1}s_1 + a_{i2}s_2 + ... + a_{in}s_n, \quad \forall i = 1,...,n \tag{1}$$

donde $\{a_{ij}\}$ con $i, j = 1,...,n$ son coeficientes reales que modelan la atenuación relativa entre las fuentes y se denominan ***coeficientes de mezcla***.

El modelo básico de ICA es un modelo generativo, en la medida en que los datos observables son generados mediante un proceso de mezcla lineal de las fuentes, siendo imposible observar estas últimas directamente. Los coeficientes de mezcla a_{ij} son desconocidos, y se deben estimar al mismo tiempo que los ***componentes independientes*** s_i a partir de los observables x_i. Este modelo básico es estático en cuanto a que las variables que intervienen en él son variables aleatorias. Omite también cualquier retardo temporal entre las fuentes que pueda darse durante la mezcla, por lo que se denomina a este modelo básico ***modelo de mezcla instantánea***.

Es conveniente utilizar una notación matricial para rescribir la ecuación (1). Sean $\underline{x}$ y $\underline{s}$ los vectores columna cuyos elementos son las mezclas $x_1,...,x_n$, y los componentes independientes $s_1,...,s_n$ respectivamente. El modelo de mezcla instantánea se escribe como:

$$\underline{x} = \underline{\underline{A}}\,\underline{s} \tag{2}$$

donde $\underline{\underline{A}}$ es la denominada matriz de mezcla formada por los coeficientes a_{ij}.

Este modelo básico no contempla la posibilidad de que el número de observables y de componentes independientes sea diferente, pero si así fuese sería importante que el número de sensores fuese superior o igual al número de fuentes para poder aplicar este método.
El objetivo del método es estimar la matriz de mezcla $\underline{\underline{A}}$ para obtener la ***matriz de separación*** $\underline{\underline{W}}$ al invertirla. Así pues se pueden recuperar los componentes independientes mediante:

$$\underline{y} = \underline{\underline{W}}\,\underline{x} \tag{3}$$

donde $\underline{y}$ contiene los componentes independientes estimados o recuperados.

Existe un gran número de criterios para estimar la matriz de separación. En cualquier caso hay varios supuestos así como dos ambigüedades sobre los componentes recuperados.

2.2 Restricciones y ambigüedades del modelo

A continuación se expresan las condiciones que debe cumplir un problema de separación ciega y las ambigüedades que se tienen al recuperar las señales.

2.2.1 Restricciones

Las suposiciones que deben hacerse para que el modelo funcione de forma correcta son las siguientes:

a. Los componentes se asumen estadísticamente independientes.

b. Los componentes, con la posible excepción de un componente, deben tener distribuciones no gaussianas.

c. Se asume el mismo número de componentes independientes que de observables, y que la matriz de mezcla es cuadrada.

2.2.2 Ambigüedades

En la estimación de los componentes independientes aparecen dos ambigüedades:
a. No queda determinada la varianza (energía) de los componentes independientes.

Los componentes estimados $\underline{y}$ son proporcionales a los originales, existe un factor de escala α_i que no se puede determinar:

$$\underline{x} = \sum_i \left(\frac{1}{\alpha_i} a_i \right) (s_i \alpha_i) \tag{5}$$

Por ello se suele asumir que los componentes a estimar tienen varianza unidad: $E\{s_i^{\,2}\} = 1$.

b. No queda determinado el orden de los componentes independientes.

Esto es debido a que se modela la mezcla mediante una matriz, y ésta no refleja el orden espacial de las fuentes originales. De forma analítica, dada una matriz de permutación $\underline{\underline{P}}$ y su inversa pueden dar $\underline{x} = \underline{\underline{A}}\,\underline{\underline{P}}^{-1}\,\underline{\underline{P}}\,\underline{s}$, donde los elementos de $\underline{\underline{P}}$ son las variables independientes originales s_j, pero en otro orden.

2.3 Preprocesado de los observables: centrado y blanqueo

Antes de realizar la separación, es conveniente realizar dos operaciones sobre los observables, que no alteran la forma de las señales, y sin embargo simplifican en gran medida el algoritmo. Este preprocesado consiste en el centrado y posterior blanqueo (incorrelación) de los datos.

2.3.1 Centrado

Con el objetivo de simplificar los algoritmos, se asume que los observables tienen media cero. En caso de no ser así, es necesario realizar un preprocesado sobre ellos para que esta suposición sea cierta. Ello es posible mediante el centrado de los observables, proceso que consiste en restarles su media $E\{\underline{x}\}$, tal como sigue:

$$\underline{x} = \underline{x}' - E\{\underline{x}'\} \quad (6)$$

Los componentes independientes estimados también tendrán media cero:

$$E\{\underline{y}\} = \underline{\underline{W}} E\{\underline{x}\} \quad (7)$$

La matriz de mezcla no cambia después de este preprocesado, por lo que siempre se podrá realizar sin que afecte a la estimación de la misma. Finalmente la estimación de los componentes independientes obtenidos a partir de los observables con media cero, deben reconstruirse sumándoles la media sustraída, $\underline{\underline{W}} E\{\underline{x}'\}$.

2.3.2 Blanqueo

El blanqueo de un vector aleatorio de media cero , $\underline{x}$, supone imponer que sus componentes estén incorrelados y que sus varianzas sean uno. En este caso la matriz de covarianza será igual a la matriz identidad:

$$\underline{\underline{C_x}} = E\{\underline{x}\underline{x}^T\} = \underline{\underline{I}} \quad (8)$$

Por tanto el blanqueado consistirá en una transformación lineal sobre los observables, multiplicándolos por una matriz $\underline{\underline{V}}$, tal que:

$$\underline{z} = \underline{\underline{V}}\,\underline{x} \quad (9)$$

$$\underline{\underline{V}} = \underline{\underline{E}}\,\underline{\underline{D}}^{-1/2}\,\underline{\underline{E}}^T = \underline{\underline{C_x}}^{-1/2} \quad (10)$$

donde el vector $\underline{z}$ tendrá covariancia diagonal, $\underline{\underline{E}}$ es la matriz ortogonal de autovectores de la matriz de covarianza y $\underline{\underline{D}}$ es la matriz diagonal de sus autovalores.

2.4 Exclusión de las variables gaussianas como fuentes

Para demostrar porqué las variables gaussianas no pueden ser componentes independientes separables se analiza qué transformación sufre la densidad de probabilidad conjunta $p_{\underline{s}}(s_1, s_2)$ de dos variables gaussianas s_1 y s_2 cuando se realiza la mezcla.

La densidad de probabilidad conjunta es:

$$p_{\underline{s}}(s_1,s_2) = \frac{1}{2\pi}\exp\left(-\frac{s_1^2+s_2^2}{2}\right) = \frac{1}{2\pi}\exp\left(-\frac{\|\underline{s}\|^2}{2}\right) \quad (11)$$

Se asume que la matriz de mezcla $\underline{\underline{A}}$ es ortogonal, dado que los datos han sido blanqueados, y por tanto $\underline{\underline{A}}^{-1} = \underline{\underline{A}}^T$, y $\underline{s} = f^{-1}(\underline{x}) = \underline{\underline{A}}^{-1}\underline{x} = \underline{\underline{A}}^T\underline{x}$. Dadas las propiedades en cuanto a las densidades de probabilidad de variables que son fruto de una transformación [7], la densidad de probabilidad conjunta de los observables x_1, x_2 viene dada por la ecuación:

$$p_{\underline{x}}(\underline{x}) = p_{\underline{x}}(x_1, x_2) = \frac{p_{\underline{s}}(f^{-1}(\underline{x}))}{\left|\det J_f(f^{-1}(\underline{x}))\right|} \quad (12)$$

$$p_{\underline{x}}(\underline{x}) = p_{\underline{x}}(x_1, x_2) = \frac{p_{\underline{s}}(\underline{\underline{A}}^T\underline{x})}{\left|\det \underline{\underline{A}}^T\right|} \quad (13)$$

$$p_{\underline{x}}(\underline{x}) = \frac{\frac{1}{2\pi}\exp\left(-\frac{\|\underline{\underline{A}}^T\underline{x}\|^2}{2}\right)}{\left|\det \underline{\underline{A}}^T\right|} \quad (14)$$

Sin pérdida de generalidad se supone que $\underline{\underline{A}}$ es ortonormal, se cumple $\|\underline{\underline{A}}^T\underline{x}\| = \|\underline{x}\|$, y puesto que $|\det \underline{\underline{A}}| = 1$, la densidad de probabilidad conjunta de los observables queda de la siguiente forma:

$$p_{\underline{x}}(\underline{x}) = \frac{1}{2\pi}\exp\left(-\frac{\|\underline{x}\|^2}{2}\right) \quad (15)$$

Se ve claramente que la transformación ortogonal no cambia la densidad de probabilidad de los datos, las distribuciones de las mezclas y de los componentes originales son idénticas. La matriz de mezcla $\underline{\underline{A}}$ no es identificable si las fuentes son gaussianas, porque en el caso de variables gaussianas conjuntas la condición de incorrelación implica necesariamente independencia.

Sin embargo sí existe la posibilidad de incluir una sola variable gaussiana en el conjunto de las fuentes. Es el caso límite en el que se pueden separar todas las fuentes si ninguna otra fuente tiene una componente gaussiana con la que pudiese estar mezclada.

3. ALGORITMOS DE GRADIENTE NATURAL EN TIEMPO Y FRECUENCIA

A continuación presentamos varios algoritmos de separación de mezclas instantáneas basados en el ***algoritmo de gradiente natural***.

En primer lugar se explica el algoritmo de máxima verosimilitud (*Maximum Likelihood ML*) [8] puesto que es una componente del algoritmo de gradiente natural, que permite calcular los coeficientes de $\underline{\underline{W}}$, es decir de la inversa de la matriz de mezcla. En segundo lugar se desarrollan dos algoritmos de gradiente, el primero de los cuales realiza la separación en el dominio temporal, mientras que el segundo lo hace en el frecuencial.

3.1 Estimación por máxima verosimilitud (ML)

Una aproximación para la estimación del modelo ICA es la estimación por máxima verosimilitud (ML). Una interpretación del estimador ML es que selecciona los valores de los parámetros que dan la probabilidad más alta para las observaciones.

La densidad $p_{\underline{x}}$ del vector de mezclas $\underline{x} = \underline{\underline{A}}\underline{s}$ se puede formular como:

$$p_{\underline{x}}(\underline{x}) = \left|\det \underline{\underline{W}}\right| p_{\underline{s}}(\underline{s}) = \left|\det \underline{\underline{W}}\right| \prod_i p_i(s_i) \quad (16)$$

donde $\underline{\underline{W}} = \underline{\underline{A}}^{-1}$, y p_i son las densidades de los componentes independientes s_i [7]. Se puede expresar en función de $\underline{\underline{W}} = (\underline{w}_1, ..., \underline{w}_n)^T$ y de $\underline{x}$, obteniendo:

$$p_{\underline{x}}(\underline{x}) = \left|\det \underline{\underline{W}}\right| \prod_i p_i(\underline{w}_i{}^T \underline{x}) \quad (17)$$

Se asume que $\underline{x}(1), \underline{x}(2), ..., \underline{x}(T)$ son T observaciones de $\underline{x}$. La verosimilitud se puede obtener como el producto de esta densidad evaluada en los T puntos. Esto se denota por L y se considera una función de $\underline{\underline{W}}$:

$$L(\underline{\underline{W}}) = \prod_{t=1}^{T} \prod_{i=1}^{n} p_i(\underline{w}_i{}^T \underline{x}(t)) \left|\det \underline{\underline{W}}\right| \quad (18)$$

Con frecuencia es más práctico trabajar con el logaritmo de la verosimilitud porque su álgebra es más sencilla y el máximo del logaritmo se obtiene en el mismo punto que el máximo de la verosimilitud. El log-verosimilitud es:

$$\log L(\underline{\underline{W}}) = \sum_{t=1}^{T} \sum_{i=1}^{n} \log p_i(\underline{w}_i{}^T \underline{x}(t)) + T \log \left|\det \underline{\underline{W}}\right| \quad (19)$$

La base del logaritmo no afecta, por lo que en lo sucesivo se hará referencia al logaritmo natural. Para simplificar la notación, se denota el sumatorio con índice t por el operador esperanza, y se divide la log-verosimilitud por T obteniendo:

$$\frac{1}{T} \log L(\underline{\underline{W}}) = E\left\{\sum_{i=1}^{n} \log p_i(\underline{w}_i{}^T \underline{x})\right\} + \log \left|\det \underline{\underline{W}}\right| \quad (20)$$

donde la esperanza se calcula como un promedio de las muestras observadas.

Existe un nuevo parámetro a estimar en el modelo ICA, las densidades de los componentes independientes, ya que la log-verosimilitud es función de ellas. Se puede usar una parametrización de p_i extremadamente simple que consiste en tomar una de las dos densidades correspondientes a dos no linealidades, detalladas más adelante, que se aplican en el método del gradiente.

3.2 Del gradiente al gradiente natural

Los algoritmos más simples para maximizar la verosimilitud se basan en calcular el gradiente de la función de coste. En este apartado se detalla como se deriva el algoritmo del gradiente natural a partir del ***algoritmo de Bell-Sejnowski*** [9].

3.2.1 Algoritmo de Bell-Sejnowski

A partir de la expresión (20), donde se realiza un promediado entre todas las muestras, se puede deducir el gradiente estocástico de la función log-verosimilitud, como:

$$\frac{1}{T} \frac{\partial \log L(\underline{\underline{W}})}{\partial \underline{\underline{W}}} = \left(\underline{\underline{W}}^T\right)^{-1} + E\left\{\underline{g}\left(\underline{\underline{W}}\underline{x}\right)\underline{x}^T\right\} \quad (21)$$

donde $\underline{\underline{W}}\underline{x} = \underline{y}$ y $\underline{g}(\underline{y}) = \begin{pmatrix} g_1(y_1) \\ \vdots \\ g_i(y_i) \\ \vdots \\ g_n(y_n) \end{pmatrix}$ es un vector de funciones $g_i(.)$, denominadas funciones *score* de las

distribuciones p_i de los componentes independientes s_i, definidas como:

$$g_i = (\log p_i)' = \frac{p_i'}{p_i} \tag{22}$$

De (21) se obtiene el siguiente algoritmo para estimación ML:

$$\underline{\underline{\Delta W}} \propto \left(\underline{\underline{W}}^T\right)^{-1} + E\left\{\underline{g}\left(\underline{\underline{W}}\,\underline{x}\right)\underline{x}^T\right\} \tag{23}$$

donde el símbolo $\propto$ indica proporcionalidad y $\underline{x} = (x_1, x_2, ..., x_n)^T$ es el vector de observables instantáneos. La versión estocástica del anterior

$$\underline{\underline{W}}\left(\underline{\underline{I}} + E\left\{\underline{g}\left(\underline{\underline{W}}\,\underline{x}\right)\underline{y}^T\right\}\right) = \underline{\underline{W}}\left(\underline{\underline{I}} + E\left\{\underline{g}\left(\underline{y}\right)\underline{y}^T\right\}\right)$$

algoritmo es el denominado algoritmo de Bell-Sejnowski:

$$\underline{\underline{\Delta W}} \propto \left(\underline{\underline{W}}^T\right)^{-1} + \underline{g}\left(\underline{\underline{W}}\,\underline{x}\right)\underline{x}^T \tag{24}$$

Dado que este algoritmo converge lentamente, es necesario, además de blanquear los datos, emplear una versión mejorada del mismo, es decir más rápida, conocida como gradiente natural o relativo. El gradiente natural además de tener mejores prestaciones en cuanto a convergencia, evita tener que invertir la matriz $\underline{\underline{W}}$.

3.2.2 Algoritmo del gradiente natural

La eficacia del gradiente natural reside en multiplicar ambos lados de la ecuación (23) por la derecha con $\underline{\underline{W}}^T\underline{\underline{W}}$, con lo que se obtiene:

$$\underline{\underline{\Delta W}} \propto \left(\left(\underline{\underline{W}}^T\right)^{-1} + E\left\{\underline{g}\left(\underline{\underline{W}}\,\underline{x}\right)\underline{x}^T\right\}\right)\underline{\underline{W}}^T\underline{\underline{W}} = \tag{25}$$

El algoritmo converge cuando $E\left\{\underline{g}\left(\underline{\underline{W}}\,\underline{x}\right)\underline{y}^T\right\} = -\underline{\underline{I}}$, es decir, cuando todos los elementos y_i y $g(y_j)$ están decorrelados para $i \neq j$.

En general las no linealidades empleadas dependen de la función de densidad de probabilidad de los componentes independientes s_i, y son:

$$f(y)^+ = -2\tanh(y) \tag{26}$$

$$f(y)^- = \tanh(y) - y \tag{27}$$

para densidades supragaussianas y subgaussianas respectivamente [3].

3.3 Algoritmos de gradiente natural para separación ciega de mezclas instantáneas

A continuación se expone la formulación teórica del algoritmo de gradiente natural tanto en el dominio temporal como en el frecuencial.

3.3.1 Algoritmo de gradiente natural en tiempo

Este primer algoritmo realiza la separación temporal de los observables blanqueados.

Sean $\underline{s}_i = (s_{i1}, s_{i2}, \cdots, s_{im})^T$, $i = 1,...,n$ los vectores de n componentes independientes de m muestras. Se define la matriz de fuentes $\underline{\underline{S}}$ tal que:

$$\underline{\underline{S}} = (\underline{s}_1, \underline{s}_2, ..., \underline{s}_n)^T \tag{28}$$

La matriz de observables $\underline{\underline{X}} = (\underline{x}_1, \underline{x}_2, ..., \underline{x}_n)^T$, donde $\underline{x}_i = (x_{i1}, x_{i2}, \cdots, x_{im})^T$, se obtiene al realizar la mezcla temporal multiplicando matricialmente:

$$\underline{\underline{X}} = \underline{\underline{A}}\,\underline{\underline{S}} \tag{29}$$

Como procesado previo a la separación, se centran y blanquean los observables, obteniendo la matriz de datos blanqueados $\underline{\underline{Z}} = \underline{\underline{V}}\,\underline{\underline{X}}$. Es conveniente trabajar con la matriz $\underline{\underline{Z}}_p$, obtenida al permutar las columnas de la matriz $\underline{\underline{Z}}$, para romper la estacionariedad de las señales.

Para un número dado de iteraciones, se calcula la matriz de separación $\underline{\underline{W}}$ inicializada a la matriz identidad antes de la primera iteración, $\underline{\underline{W}}^0 = \underline{\underline{I}}$.

En cada iteración k se estima la nueva matriz de separación con la siguiente regla de adaptación:

$$\underline{\underline{W}}^{k+1} = \underline{\underline{W}}^k + \mu.\underline{\underline{\partial W}}^k \tag{30}$$

donde μ es el denominado paso de adaptación, y $\underline{\underline{\partial W}}^k$ es el incremento matricial de la presente iteración. Este incremento se calcula mediante la expresión:

$$\underline{\underline{\partial W}}^k = \left(\underline{\underline{I}} + f(\underline{\underline{U}}^k).\frac{(\underline{\underline{U}}^k)^T}{\underline{\sigma}_{\underline{U}^k}} \right) \underline{\underline{W}}^k \qquad (31)$$

donde la matriz de componentes estimados intermedios $\underline{\underline{U}}^k$ se obtiene como:

$$\underline{\underline{U}}^k = \underline{\underline{W}}^k \underline{\underline{Z}}_p = \begin{pmatrix} \underline{u}_1^k \\ \vdots \\ \underline{u}_i^k \\ \vdots \\ \underline{u}_n^k \end{pmatrix} \qquad (32)$$

y $\underline{\sigma}_{\underline{U}^k} = \begin{pmatrix} \sigma_1^k \\ \vdots \\ \sigma_i^k \\ \vdots \\ \sigma_n^k \end{pmatrix}$ es el vector de desviaciones estándares de cada fila $\underline{u}_i^k$, y $f(\underline{\underline{U}}^k)$ es el resultado de aplicar la no linealidad $f(.)$ a la matriz $\underline{\underline{U}}^k$, en función de la naturaleza subgaussiana o supragaussiana de las señales. La normalización es necesaria para que los componentes estimados tengan varianza unidad. En caso de no tener información a priori de la naturaleza de las fuentes, es necesario evaluar en cada iteración la kurtosis de cada observable, y decidir qué no linealidad aplicar. Para las variables no gaussianas la kurtosis generalmente es diferente de cero, es negativa para las subgaussianas y positiva para las supergaussianas.

Dado que las señales de voz tienen un carácter supragaussiano (siguen una distribución próxima a la de Laplace), la no linealidad empleada puede ser $f(\underline{\underline{U}}^k) = -2\tanh(\underline{\underline{U}}^k)$.

Transcurridas todas las iteraciones, se calcula finalmente la matriz de componentes recuperados,

$$\underline{\underline{Y}} = (\underline{y}_1, \underline{y}_2, ..., \underline{y}_n)^T$$

donde,

$$\underline{y}_i = (y_{i1}, y_{i2}, \cdots, y_{im})^T$$

mediante:

$$\underline{\underline{Y}} = \underline{\underline{W}}\,\underline{\underline{Z}} \qquad (33)$$

donde $\underline{\underline{W}}$ es la matriz estimada en la última iteración.

3.3.2 Algoritmo de gradiente natural en frecuencia

Este segundo algoritmo realiza la separación frecuencial de los observables blanqueados.
Se calcula la matriz de fuentes en frecuencia $\underline{\underline{S}}^f$ aplicando la transformada de Fourier a cada uno de los vectores $\underline{s}_i$:

$$\underline{\underline{S}}^f = \begin{pmatrix} FFT(\underline{s}_1) \\ \vdots \\ FFT(\underline{s}_i) \\ \vdots \\ FFT(\underline{s}_n) \end{pmatrix} \qquad (34)$$

La matriz de observables frecuenciales,

$$\underline{\underline{X}}^f = (\underline{x}_1^f, \underline{x}_2^f, ..., \underline{x}_n^f)^T$$

donde,

$$\underline{x}_i^f = \left(x_{i1}^f, x_{i2}^f, \cdots, x_{im}^f\right)^T$$

se obtiene al multiplicar matricialmente:

$$\underline{\underline{X}}^f = \underline{\underline{A}}\,\underline{\underline{S}}^f \qquad (35)$$

En este segundo caso también se centran y blanquean los observables, obteniendo la matriz de datos blanqueados $\underline{\underline{Z}}^f = \underline{\underline{V}}\,\underline{\underline{X}}^f$.

Se calcula la matriz de separación $\underline{\underline{W}}$, donde ahora sus elementos son complejos, para un número dado de iteraciones.
En cada iteración k se estima la nueva matriz de separación con la ecuación (30), y el incremento $\underline{\underline{\partial W}}^k$ mediante la ecuación (31).

Al finalizar todas las iteraciones, se calcula la matriz de componentes frecuenciales recuperados,

$$\underline{\underline{Y}}^f = (\underline{y}_1^f, \underline{y}_2^f, ..., \underline{y}_n^f)^T$$

donde,

$$\underline{y}_i^f = \left(y_{i1}^f, y_{i2}^f, \cdots, y_{im}^f\right)^T$$

mediante la multiplicación matricial:

$$\underline{\underline{Y}}^f = \underline{\underline{W}}\,\underline{\underline{Z}}^f \qquad (36)$$

donde $\underline{\underline{W}}$ es la matriz estimada en la última iteración.

Por último, a partir de los componentes frecuenciales recuperados, se obtienen sus análogos temporales como:

$$\underline{\underline{Y}} = \begin{pmatrix} \mathrm{Re}(FFT^{-1}(\underline{y}_1^f)) \\ \vdots \\ \mathrm{Re}(FFT^{-1}(\underline{y}_i^f)) \\ \vdots \\ \mathrm{Re}(FFT^{-1}(\underline{y}_n^f)) \end{pmatrix} = \begin{pmatrix} \underline{y}_1 \\ \vdots \\ \underline{y}_i \\ \vdots \\ \underline{y}_n \end{pmatrix} \quad (37)$$

donde $\underline{y}_i = (y_{i1}, y_{12}, \text{؟}, y_{im})^T$.

4. SEPARACIÓN DE SEÑALES DE VOZ EN AMBOS DOMINIOS

Para comprobar la eficacia del algoritmo de gradiente natural, se simula una mezcla de varias voces en un coche, como si varios locutores hablaran simultáneamente en él, para poder después recuperarlas por canales separados. La separación se ejecuta tanto en tiempo como en frecuencia, mientras que la mezcla siempre se realiza en el dominio temporal.

Se realiza el análisis de la calidad de la separación de las señales de voz primero con el algoritmo en el dominio temporal y luego en el frecuencial.

Se toman como señales a mezclar seis ficheros extraídos de la base de datos SpeechDat Car Spanish propiedad de la UPC [10]. Los ficheros de referencia seleccionados para las mezclas, así como su contenido son:

a. V10000D1: *"siete de octubre del setenta y seis"*,
b. V10010A1: *"finalizar la llamada"*,
c. V10021T1: *"son las cuatro y cinco"*,
d. V10020O5:*"avenida del puerto"*,
e. V10011C4: *"0 9 5 0 9 6"*, y
f. V1003051: *"lista de destinos"*.

4.1. Separación en el dominio temporal

Para hacer una separación de señales temporales, se carga la matriz de mezcla que deseemos aplicar al sistema. Una vez comprobada si es válida, es decir si es invertible, se realiza la mezcla de las fuentes, y se obtienen de forma matricial las fuentes mezcladas, es decir los observables.

Se realizan dos medidas distintas de calidad de la separación, una vez obtenida la matriz de separación. La primera es el cálculo de la evolución de las relaciones señal a error, es decir la relación entre la potencia de cada señal original, con respecto a la potencia del error entre dicha señal y su recuperada. Este parámetro se denomina SER, y se define como:

$$SER_i = \frac{1/N \sum_{j=1}^{N} (s_i^j)^2}{1/N \sum_{j=1}^{N} (s_i^j - y_i^j)^2}, \; i = 1,2,3$$

$$j = 1,\ldots,N \quad (38)$$

donde N es el número de muestras de las señales original y recuperada. Se calcula esta medida en decibelios, para cada una de las señales de voz recuperadas. También se recoge la media de las mismas.

La segunda medida de calidad consiste en el cálculo del valor de la kurtosis de las señales de voz originales y recuperadas, así como el error entre ambas. La kurtosis de una variable aleatoria x, que por simplicidad asumiremos de media cero y varianza igual a uno, viene dada por:

$$kurt(x) = E\{x^4\} - 3\left(E\{x^2\}\right)^2 \quad (39)$$

y dado que $E\{x^2\} = 1$ se tiene que $kurt(x) = E\{x^4\} - 3$.

La kurtosis es una medida de parecido entre una distribución dada y la distribución gaussiana para la cual la kurtosis es igual a cero. Así, cuanto mayor es el valor absoluto de la kurtosis, menos se parece la distribución a una gaussiana.

Se muestran resultados para una mezcla de los tres primeros ficheros mencionados, un paso de adaptación de 0.01 y una matriz de mezcla simple

$$\underline{\underline{A}} = \begin{bmatrix} 1 & 2 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Se realiza una simulación de 1000 iteraciones. El algoritmo converge desde la primera iteración a los siguientes valores:

$$SER_1 = 14{,}2981\,\mathrm{dB}$$
$$SER_2 = 17{,}5244\,\mathrm{dB}$$
$$SER_3 = 11{,}2809\,\mathrm{dB}$$

Es evidente que si las señales recuperadas fuesen idénticas a las originales, el valor del parámetro sería infinito. Al realizar la media aritmética del error cuadrático de cada muestra, se obtiene un valor

ponderado del mismo. Los valores obtenidos son por tanto relativamente buenos, puesto que en todo caso la señal recuperada tiene una potencia más de diez veces mayor a la potencia de la señal error. La evolución de este parámetro para cada una de las señales de voz, así como la media de las tres puede verse en la Figura 1.

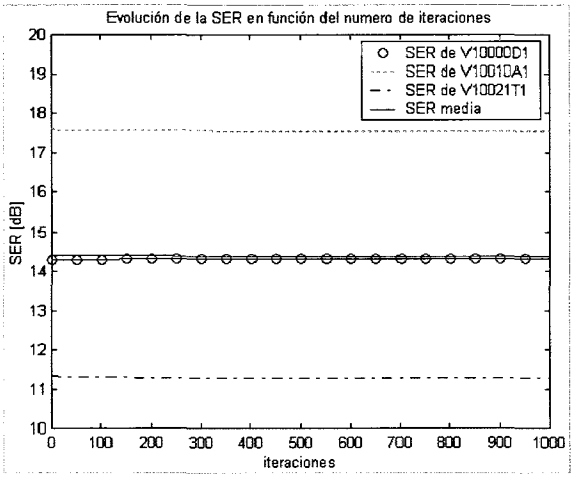

Figura 1: *Evolución de las SER en la separación temporal de tres señales de voz.*

Se aprecia claramente que la SER varía muy poco desde la primera iteración, reafirmando el hecho de que el algoritmo del gradiente natural converge de forma rápida.

La Tabla 1 recoge los valores de las kurtosis de las señales originales, recuperadas y error. Se observa que las kurtosis de las señales originales y recuperadas son casi idénticas, como lo demuestra el bajo error entre ambas.

Fichero	Kurtosis de la señal original	Kurtosis de la señal recuperada	Error entre las kurtosis
V10000D1	11,1493	11,1504	0,0011
V10010A1	21,5416	21,5601	0,0185
V10021T1	17,3213	17,3572	0,0359

Tabla 1: *Comparación de las kurtosis de la separación temporal de tres señales.*

4.2 Separación en el dominio frecuencial

Para ver cómo se comporta el algoritmo de gradiente natural al separar señales frecuenciales, se utiliza un procedimiento análogo al anterior. Cabe mencionar que antes de realizarse la mezcla se pasa al dominio frecuencial, para posteriormente hacer la separación, y finalmente volver al dominio temporal, en el que se realizan las medidas de calidad, que son las mismas que anteriormente.

Con el objetivo de ver la evolución del algoritmo en función del número de iteraciones, se realiza una simulación para 1000 iteraciones. La evolución de la SER en este caso se muestra en la Figura 2, donde puede observarse como las evoluciones de las SER son crecientes hasta que alcanzan su valor óptimo y empiezan a decrecer, hasta que sus valores se estabilizan. El número óptimo de iteraciones es aquel que maximiza las SER y minimiza el error entre las kurtosis. Este número es diferente para cada SER, por tanto se considerará el valor óptimo de la SER media para determinarlo. Siguiendo este criterio y tal como puede observarse en la Figura 2, el número óptimo de iteraciones es 85.

Los valores óptimos de las diferentes SER recogidos de la iteración 85 son:

$$SER_1 = 47{,}4071\,\text{dB}$$

$$SER_2 = 54{,}4948\,\text{dB}$$

$$SER_3 = 33{,}5186\,\text{dB}$$

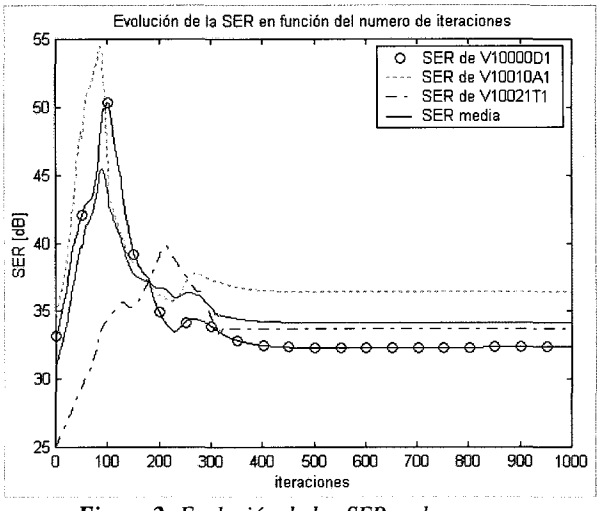

Figura 2: *Evolución de las SER en la separación frecuencial de tres señales de voz.*

Fichero	Kurtosis de la señal original	Kurtosis de la señal recuperada	Error entre las kurtosis
V10000D1	11,1493	11,1486	0,0008
V10010A1	21,5416	21,5435	0,0019
V10021T1	17,3213	17,3477	0,0265

Tabla 2: *Comparación de las kurtosis de la separación frecuencial de tres señales*

Los valores obtenidos en frecuencia, para los mismos parámetros de entrada, son ampliamente superiores en

este segundo caso. La segunda señal de voz sigue siendo la que se recupera con mayor calidad de las tres.

La Tabla 2 muestra como las kurtosis de la señal original y recuperada llegan a parecerse mucho en la iteración 85, siendo el error entre ellas muy próximo a cero.
En segundo lugar se mezclan los seis ficheros de referencia con la matriz:

$$\underline{\underline{A}} = \begin{bmatrix} 1 & 2 & 0{,}02 & 0{,}1 & 1 & 0{,}01 \\ 1 & 1 & 0{,}3 & 2 & 1 & 0{,}5 \\ 1 & 0{,}2 & 1 & 0{,}135 & 1 & 1 \\ 0{,}15 & 0{,}01 & 2 & 1 & 0{,}02 & 1 \\ 0{,}2 & 1 & 0{,}1 & 1 & 0{,}02 & 1 \\ 0{,}03 & 2 & 0{,}2 & 1 & 1{,}5 & 0{,}1 \end{bmatrix}$$

y se emplea el mismo paso de adaptación. Cabe mencionar que la elección del paso de adaptación influye en la velocidad de convergencia, y que implica emplear un número mayor o menor de iteraciones para llegar al punto óptimo. Tras 1300 iteraciones las SER de cada señal ha sufrido una

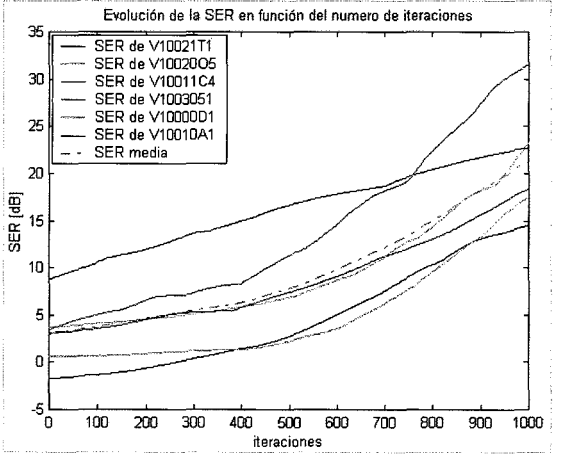

Figura 3: *Evolución de las SER en la separación frecuencial de seis señales de voz.*

importante evolución hasta estabilizarse como puede verse en la Figura 3.

Los valores óptimos de las diferentes SER recogidos de la iteración 1300 son:

$SER_1 = 21{,}9222\text{dB}$

$SER_2 = 23{,}5876\,\text{dB}$

$SER_3 = 15{,}2672\text{dB}$

$SER_4 = 29{,}9398\,\text{dB}$

$SER_5 = 24{,}2032\text{dB}$

$SER_6 = 32{,}7914\,\text{dB}$

Estos valores son inferiores a los obtenidos en el caso anterior. Con la incorporación a la mezcla de otras tres señales, la calidad de las tres primeras empeora levemente.
La Tabla 3 recoge las medidas de kurtosis para este último caso.

Fichero	Kurtosis de la señal original	Kurtosis de la señal recuperada	Error entre las kurtosis
V10000D1	11,1493	11,0561	0,0932
V10010A1	21,5416	21,5172	0,0245
V10021T1	17,3213	17,1251	0,1962
V10020O5	9,7846	9,5673	0,2173
V10011C4	8,9862	8,9831	0,0031
V1003051	21,2508	21,1885	0,0623

Tabla 3: *Comparación de las kurtosis de la separación frecuencial de seis señales.*

Los valores de error entre las kurtosis han aumentado, pero siguen siendo muy bajos.

Una mayor complejidad de la mezcla, así como un mayor número de fuentes afectan moderadamente a la calidad de la separación. Sin embargo auditivamente las señales recuperadas se corresponden sin confusión alguna con las originales.

5. CONCLUSIONES

Una vez expuestos los fundamentos del método de Análisis de Componentes Independientes se han presentado las bases del algoritmo de gradiente natural. Se han desarrollado dos algoritmos de gradiente para la separación ciega de fuentes de voz, uno en tiempo y el otro en frecuencia.

Los índices de calidad obtenidos mediante la separación en el dominio frecuencial son superiores a los obtenidos en el temporal. En este caso las señales tienen siempre una potencia de error menor, confirmando así una mayor calidad de la separación.
Además en este caso sí que existe una importante mejora de la relación señal ruido segmental al aumentar el número de iteraciones. Por tanto si la separación se realiza en el dominio frecuencial, el número de iteraciones se convierte en un factor fundamental en la eficacia del algoritmo.

Como conclusión a este estudio, se puede destacar que las dos medidas de calidad de la separación (cálculo de la relación señal a error y comparación de las kurtosis de las señales originales y recuperadas) realizada mediante el algoritmo del gradiente natural, revelan la eficacia del mismo. Cabe mencionar que dicho algoritmo funciona correctamente, es decir que separa auditivamente las voces mezcladas, desde la primera iteración. Con el incremento del número de iteraciones en el caso frecuencial, se dan un aumento de la relación señal a error y la disminución de la diferencia entre las kurtosis de las señales.

Estos hechos confirman la idoneidad de realizar la separación ciega de mezclas instantáneas con algoritmos de gradiente que trabajen en el dominio frecuencial.

BIBLIOGRAFÍA

[1] A. Prieto, B. Prieto, C.G. Puntonet, A. Cañas, P. Martín-Smith. *«Geometric separation of linear mixtures of sources: Application to speech signals»*. International workshop on Independent Component Analysis and Blind Signal Separation (ICA'99) , pp. 295-300, Aussois, France, Enero 11-15, 1999.

[2] A. Westner, V.M. Bove. *"Blind separation of real world audio signals using overdetermined mixtures"*. Proc of ICA'99, pp. 251-256, Enero 11-15, Aussois, Francia, 1999.

[3] Aapo Hyvärinen, Juha Karhunen, Erkki Oja. *Independent Component Analysis.* Wiley Interscience 2001.

[4] Te-Won Lee. *Independent Component Analisys, theory and applications.* Kluwer Academic Publishers.

[5] Aapo Hyvärinen. *"Survey on Independent Component Analysis"*. Neural Computing Surveys 2, pp. 94-128,1999.

[6] Te-Won Lee, Mark Girolami, Anthony J. Bell, Terrence J. Sejnowski. *"A Unifying Information-theoretic Framework for Independent Component Analysis"*. International Journal on Mathematical and Computer Modeling.

[7] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 3erd edition, 1991.

[8] B.A. Pearlmutter, L. C. Parrra, *"Maximum Likelihood Blind Source Separation: A Context-Sensitive generealization of ICA"*. Advances in Neural Information Processing Systems. 1997.

[9] A.J. Bell, T.J.Sejnowski. *"An information-maximization approach to blind separation and blind deconvolution"*. Neural Computation 7, pp.1129-1159, 1995.

[10]A. Moreno. *"Documentación de la base de datos SpeechDat Car Spanish"*. V2. Universitat Politècnica de Catalunya. 12 Septiembre 1999.

AUTORES

Irene Aldecoa Bilbao.
Nació en Barcelona el 3 de septiembre de 1979. Estudió Ingeniería Superior de Telecomunicaciones en Barcelona. En la actualidad está realizando el proyecto fin de carrera en el departamento de Teoría de Señal y Comunicaciones de la ETSETB.

Noemí Carné Serrano .
Nació en Barcelona el 12 de junio de 1979. Estudió Ingeniería Superior de Telecomunicaciones en Barcelona. En la actualidad está realizando el proyecto fin de carrera en el departamento de Teoría de Señal y Comunicaciones de la ETSETB.

Enric Monte Moreno.
Se graduó y se doctoró en Ingeniería de Telecomunicación por la Universidad Politécnica de Cataluña (UPC) en 1987 y 1992 respectivamente. Vinculado al departamento de Teoría de Señal y Comunicaciones desde el año 1989, actualmente ejerce el cargo de profesor titular. Su interés actual se centra en el tratamiento de señal de voz.